

1/7

FIG. 1

$K = 3, l = 2$  convolutional encoder

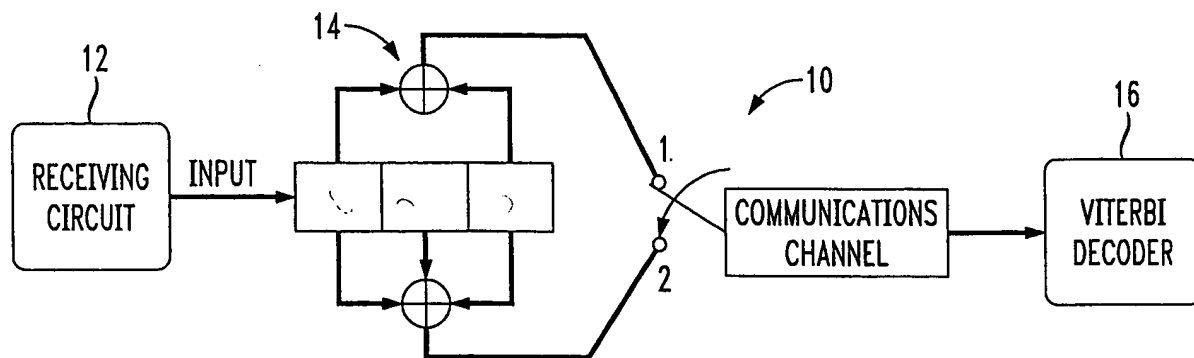
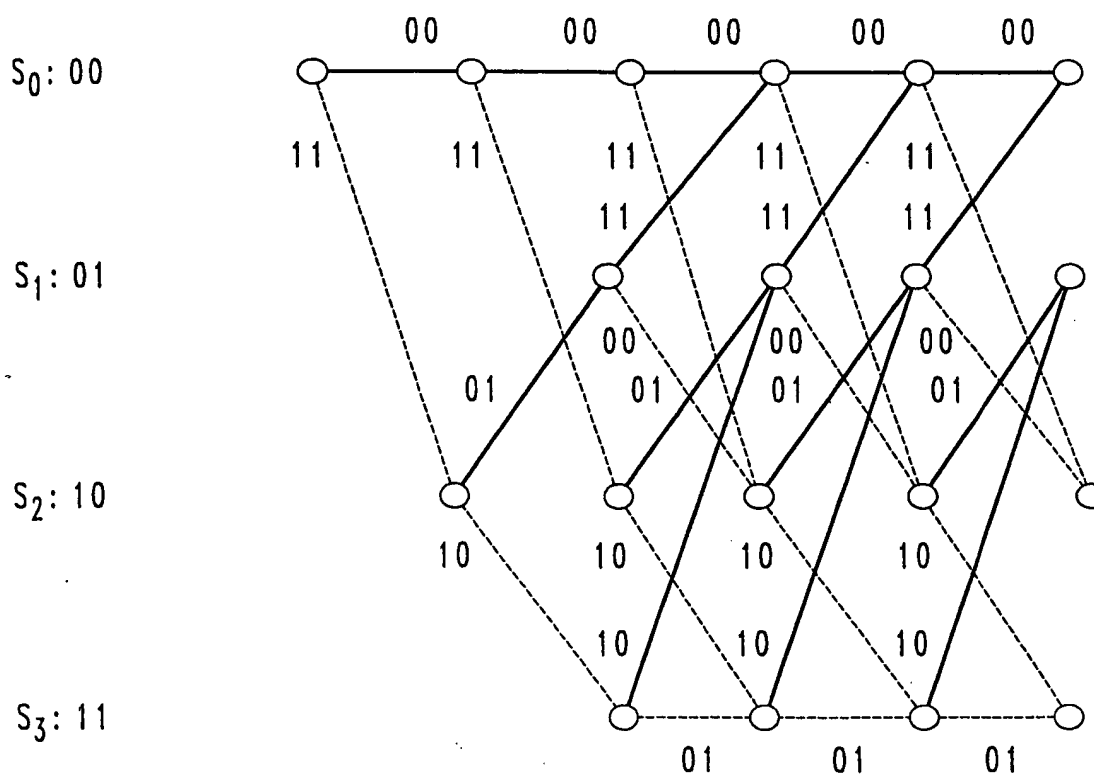


FIG. 2

Trellis for ordinary convolutional code





2/7

FIG. 3

Inserting zero at the first position periodically

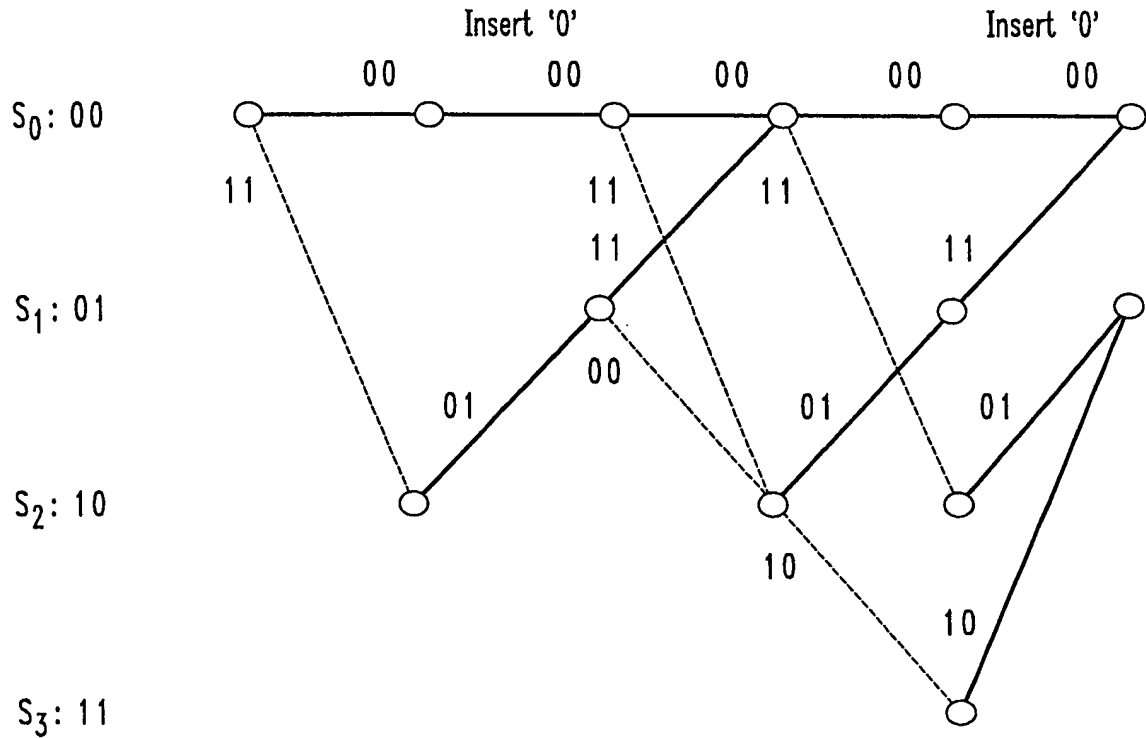
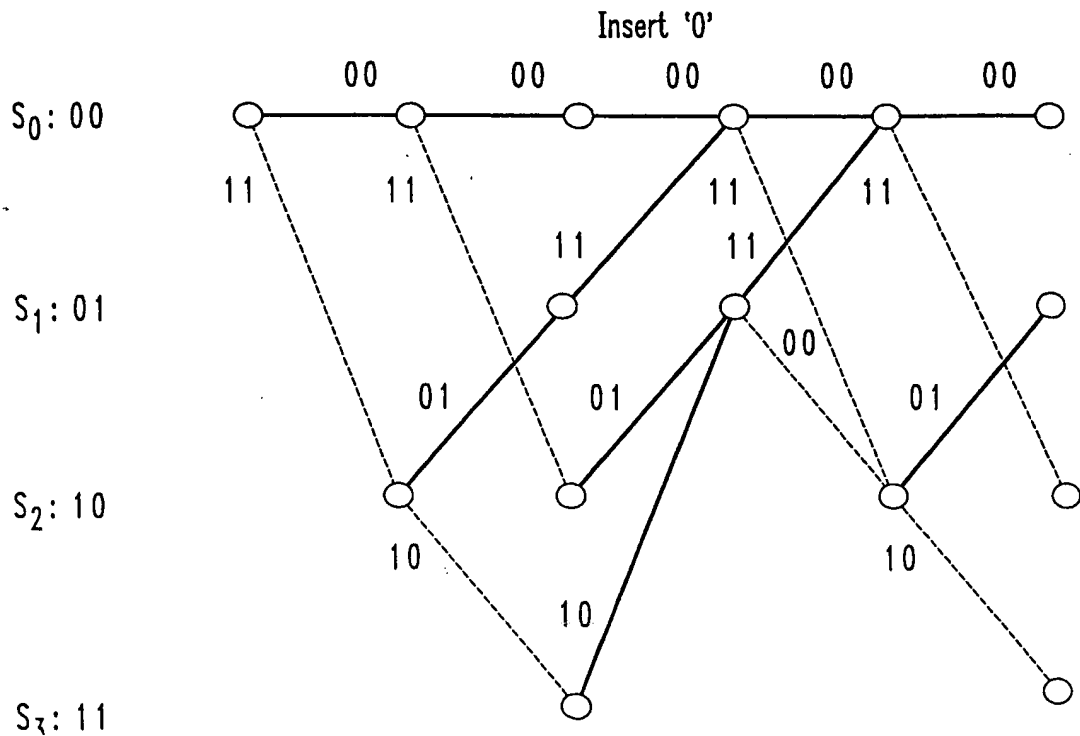
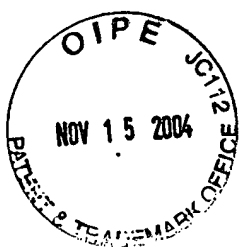


FIG. 4

Insert zero at the second position periodically





3/7

FIG. 5

Generator Matrix

Let

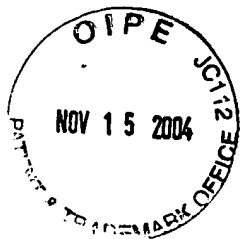
$$C(j) = X(j)G, \quad j = 1, 2, \dots, K-1, \quad (1)$$

where  $X(j) = [1, x_1, \dots, x_{j-1}, 0, x_{j+1}, \dots]$ ,  $x_{tK+j} = 0, t = 0, 1, \dots$ ,  $G$  is the Toeplitz block matrix

$$G = [\vec{g}_{i-j}]_{i,j=0,1,\dots}$$

with  $1 \times K$  sub-blocks

$$\vec{g}_i = \begin{cases} [g_{1,i}, g_{2,i}, \dots, g_{l,i}], & i = 0, 1, \dots, m; \\ 0, & \text{others.} \end{cases}$$



4/7

*FIG. 6*

Gj Presentation

$$\begin{bmatrix} \vec{g}_0(t) & \vec{g}_1(t+1) & \cdots & \vec{g}_m(t+m) & \cdots \\ 0 & \vec{g}_0(t+1) & \cdots & \vec{g}_{m-1}(t+m) & \cdots \\ . & . & \cdots & . & \cdots \end{bmatrix},$$

*FIG. 6A*

$$\begin{aligned} & \phi_t \left( X_{t-K+2}^t \right) \\ = & \max_{X_0^{t-K+1}} M \left( X_0^t \right) \\ = & \max_{x_{t-K+1}} \left[ L \left( X_{t-K+1}^t \right) + \phi_{t-1} \left( X_{t-K+1}^{t-1} \right) \right] \end{aligned}$$



5/7

**FIG. 7A**

DECODING

**Step 1 Initialization:** For  $0 \leq t < K - 1$ , starting from  $\phi(X_{-K}^{-1}) = 0$  we calculate  $\phi(X_{t-K+1}^t)$  for all possible combinations of  $X_0^t$  by (3).

30

**Step 2 Recursive forward algorithm at  $t$ :**  
If  $t \neq K - 1 \pmod{K}$ , we compute  $\phi(X_{t-K+2}^t)$  by (3) and save

$$\begin{aligned} & \tilde{x}_{t-K+1}(X_{t-K+2}^t) \\ &= \arg \max_{x_{t-K+1}} \left[ L(X_{t-K+1}^t) + \phi(X_{t-K+1}^{t-1}) \right]; (5) \end{aligned}$$

otherwise we compute  $\phi(X_{t-K+2}^t)$  by (4).

Go to Step 3.

32

STEP 3



6/7

FIG. 7B

**Step 3 Recursive backward algorithm at  $t$**

If  $t - D \neq K - 1 \pmod{K}$ , starting from

$$\hat{X}_{t-K+2}^t = \arg \max_{X_{t-K+2}^t} \phi \left( X_{t-K+2}^t \right) \quad (6)$$

we calculate  $\hat{x}_k = \tilde{x}_k \left( \hat{X}_{k+1}^{k+K-1} \right)$ ,  $k = t - K + 1, t - K, t - K - 1, \dots$  until backward  $D$  symbols to find

$$\hat{x}_{t-D} = \tilde{x}_{t-D} \left( \hat{X}_{t-D+1}^{t-D+K-1} \right); \quad (7)$$

otherwise  $\hat{x}_{t-D} = 0$ .

$T \neq N$ , Back to Step 2 34

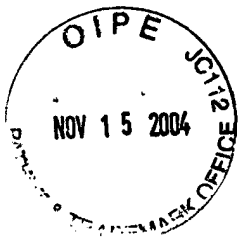
If  $t = n$  go to Step 4; otherwise go to Step 2. 36

**Step 4 Termination:** Let  $n \leq t < n + K - 2 = N$ .

If  $t \neq K - 1 \pmod{K}$ , we compute  $\phi \left( X_{t-K+2}^t \right)$  by (3) and save  $\tilde{x}_{t-K+1} \left( X_{t-K+2}^t \right)$  by (5); otherwise we compute  $\phi \left( X_{t-K+2}^t \right)$  by (4) and we do not need to save  $\tilde{x}_{t-K+1} \left( X_{t-K+2}^t \right)$  since it must be zero.

Repeat this step until  $t = N$ , then go to Step 5.

To Step 5 38



7/7

FIG. 7C

From Step 4

40

**Step 5 Recursive backward algorithm at the end:** Starting from

$$\hat{x}_n = \arg \max_{x_n} \phi \left( \underbrace{0, \dots, 0}_{K-2}, x_n \right),$$

we estimate  $x_t$  by

$$\hat{x}_t = \tilde{x}_t \left( \hat{X}_t^{t+K-2} \right), \quad t = n-1, n-2, \dots, n-D.$$

FIG. 8

Code	Conv. Code	Conv. Zero Code
Code Rate	$\frac{T}{(T+K-1)l} \approx \frac{1}{l}$	$\frac{T}{Nl} \approx \frac{K-1}{Kl}$
Complexity	$\approx T(l+2)2^K$	$\approx \frac{K}{K-1} T(l+2)2^{K-1}$
Memory	$2^{KD}$	$2^{K-1} \left( D - \left\lceil \frac{D}{K} \right\rceil \right)$
Delay	$D$	$D$